



**Instituto Federal de Educação, Ciência e Tecnologia da Paraíba**

**Professor:** Alexandre Sales Vasconcelos

**Alunos:** Matheus Alves da Silva e Hércules de Sousa Silva

**Disciplina:** Sistemas Embarcados

**Curso:** Engenharia de Computação

## **Projeto Caixa de Água**

Campina Grande — PB

2023

## **1 Introdução**

Um sistema de monitoramento e controle de temperatura e nível de água de um reservatório é um projeto importante para várias aplicações. O controle de temperatura é utilizado em muitos contextos para garantir a segurança e a qualidade da água armazenada, pois, temperaturas inadequadas podem alterar o sabor e o odor da água ao permitir a proliferação de bactérias e outros microrganismos. Já no que diz respeito ao controle de nível, ele se faz necessário para garantir que o reservatório esteja sempre abastecido, o que pode ser muito útil para aplicações que não podem ficar água, como sistemas de irrigação ou refrigeração, processos industriais, etc. Portanto, este produto é uma ferramenta importante para garantir segurança, qualidade e eficiência em muitos contextos que dependem do correto armazenamento de água.

## **2 Objetivo geral**

Este projeto tem como objetivo a construção de um sistema de controle e monitoramento de água e temperatura utilizando o microcontrolador ESP-32, a fim de colocar em práticas os conhecimentos obtidos na disciplina de sistemas embarcados ao desenvolver um sistema capaz de aquecer a água de um reservatório e de garantir que ele esteja sempre cheio.

## **3 Objetivo específicos**

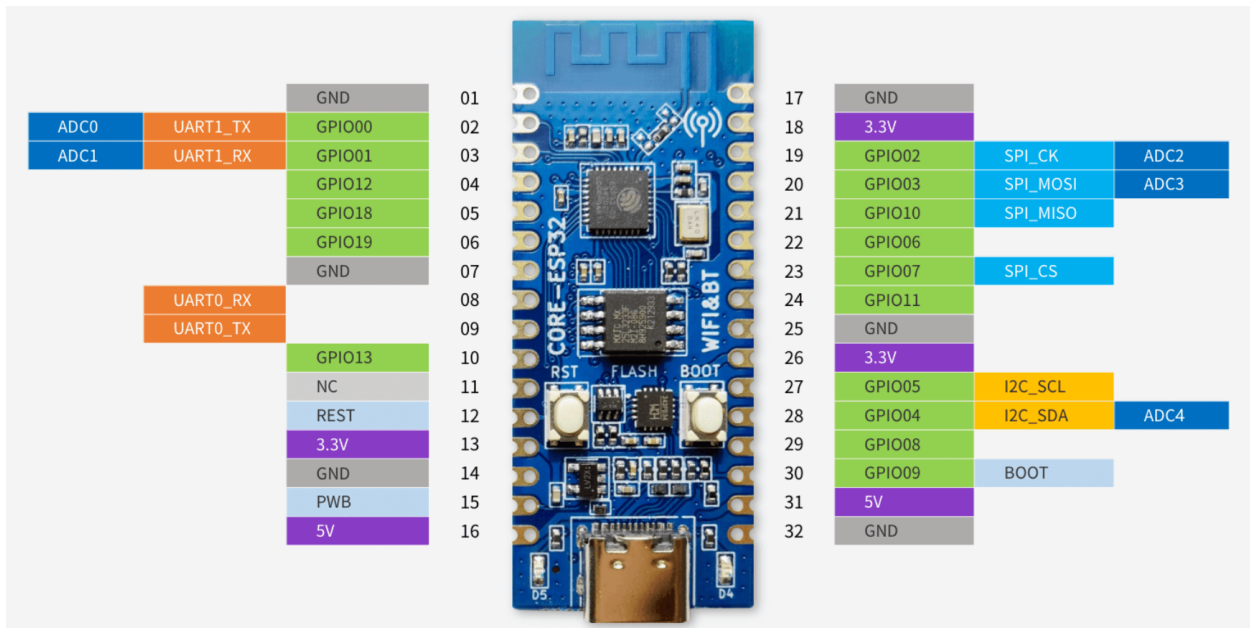
O sistema a ser desenvolvido deve ser capaz de:

- Monitorar a temperatura de um reservatório;
- Aquecer a água presente na caixa;
- Monitorar o nível de água;
- Controlar uma bomba para encher o reservatório;
- Exibir em um display informações sobre o sistema sendo monitorado;
- Diminuir o desperdício de água.

## 4 Materiais e métodos

### I.

Na construção desse projeto foi utilizado um microcontrolador ESP32 modelo C3 para o gerenciamento das funcionalidades do sistema, este foi conectado a uma protoboard como placa de ensaio onde pode-se testar as conexões com os demais elementos do circuito. Para a captação de informações referentes ao nível do reservatório foi usado o sensor ultrassônico HCSR04 que funciona com base no princípio de ecolocalização, onde pulsos ultrassônicos são emitidos, navegam pelo ambiente, atingem algum objeto e são refletidos de volta para o sensor que através do tempo de transmissão e retorno é capaz de calcular a distância entre a origem do pulso e o objeto. Para realizar a integração do sensor de distância e a ESP 32 foi preciso procurar no datasheet do microcontrolador por pinos que tivessem função de multiplexação UART, os pinos escolhidos com base nessa busca foram o GPIO 0 e o 1, sendo o primeiro do tipo TX, portanto conectado ao pino TRIGGER do sensor que é o responsável por emitir o pulso ultrassônico, e o segundo que é do tipo RX foi conectado ao pino ECHO do sensor que contém o eco da pulsação emitida. Segue abaixo a imagem de descrição de cada um dos pinos da ESP32 C3:



Para o monitoramento da temperatura do reservatório foi utilizado o sensor de temperatura DS18B20. A entrada de dados do sensor de temperatura foi conectada ao GPIO 13 da ESP32 passando através de um resistor de pull-up de  $4.7K\Omega$  que também foi colocado entre o sinal VCC conforme descrito no datasheet do sensor.

## II.

- a. um microcontrolador ESP32 modelo C3;



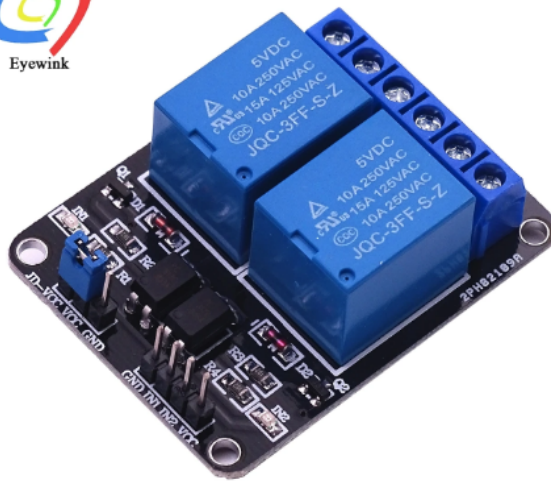
- b. Um sensor de distância ultrassônico HC SR04;



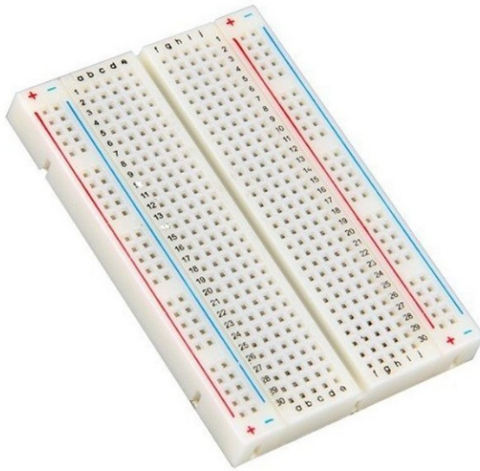
- c. Um sensor de temperatura DS18B20;



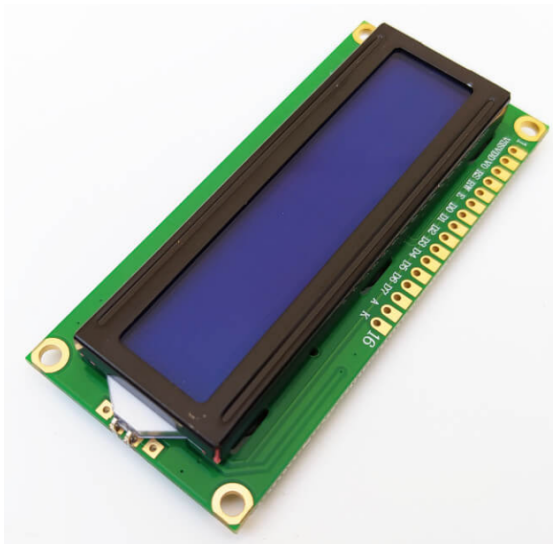
d. Um relé de dois canais;



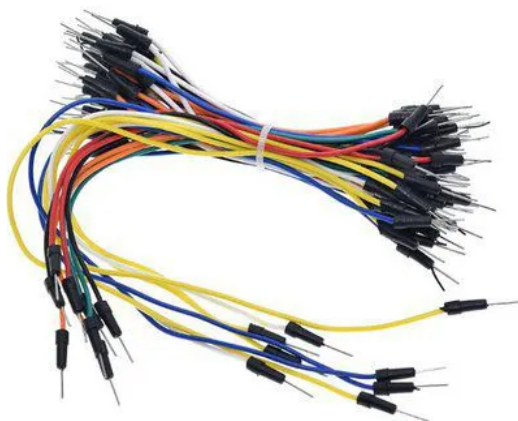
e. Uma protoboard;



f. Um display;



g. Jumpers;



#### h. Resistências.



### III.

As bibliotecas que foram para o desenvolvimento desta aplicação foram:

- **stdio:** biblioteca padrão da linguagem de programação C (e também em C++) que fornece funções e macros para entrada e saída de dados. Ela permite que você interaja com dispositivos de entrada/saída, como o teclado e o monitor, facilitando a leitura e escrita de dados em um programa;
- **freertos/freeRTOS:** essa biblioteca fornece um ambiente de execução multitarefa e permite a criação de aplicativos que requerem escalonamento de tarefas, sincronização, gerenciamento de memória e comunicação entre tarefas;

- freertos/task: o módulo task da biblioteca freertos é responsável pela criação, gerenciamento e controle das tarefas no sistema operacional FreeRTOS. Esse módulo fornece funções e macros para trabalhar com tarefas e permite que você crie e controle a execução de várias tarefas no seu programa;
- esp\_system: essa biblioteca fornece funções e macros relacionadas ao sistema e ao gerenciamento de energia de microcontroladores;
- driver/gpio: esse módulo é usado para controlar as operações de Entrada/Saída de Propósito Geral (GPIO);
- esp\_log: essa biblioteca é usada para fazer o registro de mensagens (logs) durante a execução de um programa em um microcontrolador;
- esp\_timer: essa biblioteca é utilizada para criar e controlar timers (temporizadores) nos microcontroladores;
- esp\_err: é usada para representar e gerenciar erros em operações nos microcontroladores;
- ds18x20: é uma biblioteca de software que fornece funções e recursos para facilitar o uso dos sensores de temperatura da família DS18x20, como o DS18B20;
- inttypes: é uma biblioteca em linguagem C que faz parte do padrão C99 e é usada para manipulação de tipos de dados inteiros com tamanho específico e portabilidade entre plataformas